



Høgskolen i Telemark

Avdeling for allmenne fag

EKSAMEN

I

801 OBJEKTBASERT PROGRAMMERING

14.05.2002

Tid: 09:00-14:00

Målform: Bokmål / nynorsk

Sidetal: 11 inkludert framsida

Hjelpemiddel: Alle skriftlege

Merknader: Ingen

Vedlegg: Ingen



Bokmål

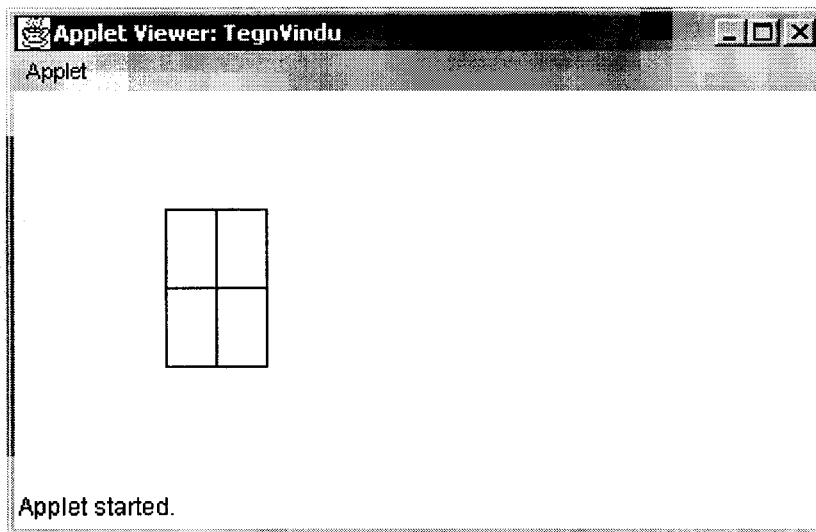
Oppgavesettet inneholder tre uavhengige deler. Prosentsatsen ut for hver oppgave antyder vekt ved sensur. Alle deloppgavene i en gitt oppgave teller like mye. Legg vekt på å besvare så mange oppgaver som mulig. Lykke til!

Oppgave 1: Grafikk og metoder (40%)

I denne oppgaven skal du skrive *metoder*, ikke komplette programmer.

1-a)

Lag en metode *tegnVindu* som tegner et vindu med to sprosser, slik figuren viser:

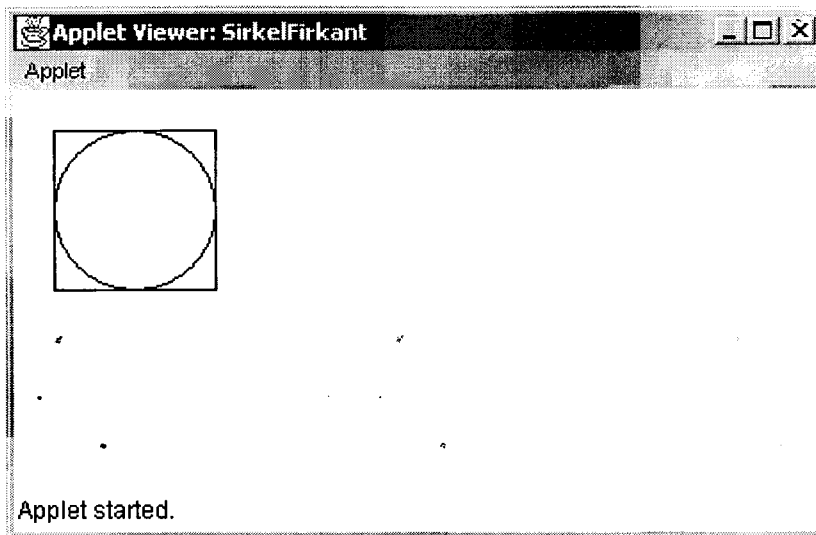


```
public void tegnVindu( Graphics g, int midtX, int midtY, int  
    bredde, int høyde){  
    // Skriv kode her.  
}
```

der midtX og midtY er koordinatene til punktet der srossene møtes midt i vinduet, og bredde og høyde er h.h.v. vinduets bredde og høyde.

1-b)

Lag en metode *tegnSirkelFirkant* som tegner en sirkel omsluttet av en firkant, slik figuren viser:

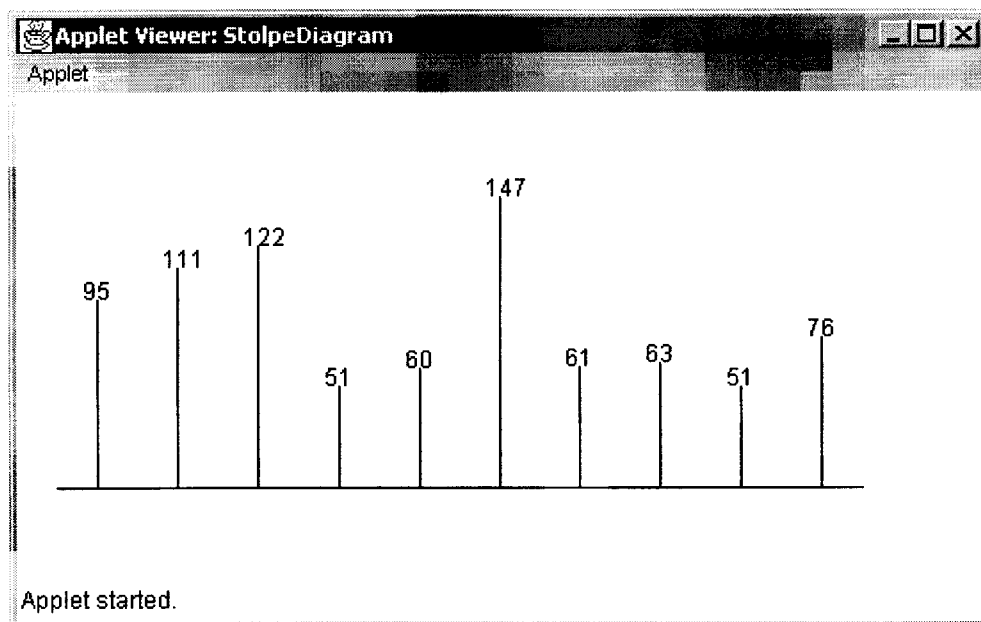


```
public void tegnSirkelFirkant ( Graphics g, int midtX, int  
midtY, int radius) {  
    // Skriv kode her.  
}
```

der midtX, midtY er koordinatene til sirkelens senter, og radius er radien til sirkelen.

1-c)

Lag en metode *tegnStolpeDiagram* som tegner et stolpediagram, slik figuren viser:



```
public void tegnStolpediagram( Graphics g, int maxY, int antallStolper ){  
    // Skriv kode her.  
}
```



Den første stolpen tegnes 40 piksler inn i x-retning. Avstanden mellom hver stolpe er 40 piksler, høyden til hver stolpe er et tilfeldig tall mellom grensene 0 og maxY. Metoden genererer selv stolpehøyden (tallene i figuren over er bare et eksempel). Det skal tegnes antallStolper stolper. Tegn x-aksen slik at den begynner 20 piksler inn og (maxY +50) piksler ned i appletvinduet, og har en lengde som er 40 ganger antall stolper.

1-d)

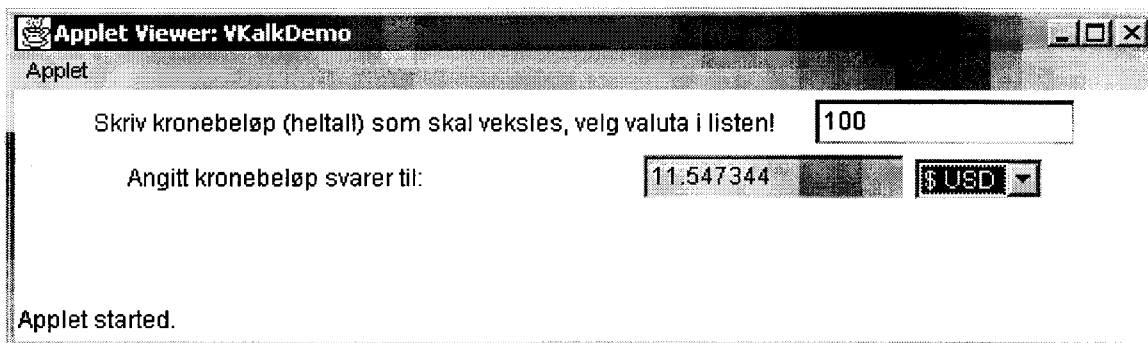
Skriv en metode *fillMedPartall* som oppretter, initialiserer og returnerer en tabell. Tabellen skal fylles med alle partallene (i stigende rekkefølge) fra 2 til det partallet som er mindre eller lik parameteren n.

Eks: metodekallet `fillMedPartall(9)` returnerer en referanse til tabellen {2, 4, 6, 8}.

```
public int[] fillMedPartall ( int n ) {  
    // Skriv kode her.  
}
```

Oppgave 2: Valutakalkulator (30%)

I denne oppgaven skal du lage en applett med tilhørende modellklasse (model-view-controller). Appletten skal ha følgende brukergrensesnitt:



- Brukeren skal skrive inn et kronebeløp i et tekstfelt (kun heltall), og velge en valuta fra en dropdown liste (klasse Choice).
- Programmet skal da regne ut hva kronebeløpet svarer til i valgt valuta, og vise frem resultatet i et tekstfelt, som vist i figuren.

Tips: Klassen Choice har en metode `public int getSelectedIndex()` som returnerer indeksen til det valgte elementet i Choice-listen.

For enkelhets skyld er variablene *valuta* og *valutakurs* gitt direkte i modellklassen VKalk, som er skissert under. Det er IKKE forklart hva de ulike metodene skal gjøre, det må du selv finne ut.

```
public class VKalk{
```



```
private String[] valuta = { "NOK", "£ GBP", "$ USD", "Euro", "Yen" };
private float[] valutaKurs = {1.0f, 12.447f, 8.660f, 7.623f, 6.556f };
// andre instansvariabler i VKalk

public VKalk(){
    // kode for konstruktør
}
public int getValutaAntall(){
    // Skriv kode her.
}
public String getValuta( int index){
    // Skriv kode her.
}
public void setBeløp( float Beløp ){
    // Skriv kode her.
}

public float getValutaBeløp(){
    // Skriv kode her.
}

public void setValutaIndex( int index){
    // Skriv kode her.
}

// Hvis du trenger hjelpemetoder kan du lage det
}
```

2a)

Skriv kode for appleten VKalkDemo. Appleten skal bruke metoder i modellklassen VKalk. Det kan lønne seg å studere modellklassen godt før du begynner å programmere.

2b)

Bestem nødvendige instansvariabler og skriv kode til alle metodene i modellklassen VKalk.

Oppgave 3: Ikke-grafisk applikasjon med filbehandling (30%)

En palindrom setning er en setning der bokstavenes rekkefølge er den samme når setningen leses både forlengs og baklengs. Alle tegn som ikke er bokstaver (komma, mellomrom, utropstegn, spørsmålstegn osv) ignoreres i denne sammenheng. Om det er store eller små bokstaver ignoreres også.

Her er noen eksempler (på engelsk):

Dammit, I'm mad!

A nut for a jar of tuna.

A santa lived as a devil at NASA.

A Toyota.

Flee to me, remote elf!



GNU dung.

Kayak.

Name no one man,

"Naomi", I moan.

Yawn--Madonna fan? No Damn way!

Lag en ikke-grafisk applikasjon som leter gjennom en tekstfil etter setninger som er palindrome. Hver setning står på en egen linje. Programmet oppretter en ny fil og skriver eventuelle palindrome setninger til denne nye filen.

Her er en hjelpemetode du kan bruke i løsningen din:

```
private String fjernSpesialTegn( String linje){  
    String bokstaver = "";  
    for (int i = 0; i < linje.length(); i++){  
        char tegn = linje.charAt(i);  
        if (Character.isLetter(tegn) )  
            bokstaver+=tegn;  
    }  
    return bokstaver;  
}
```

Programmet startes slik:

```
C:\ java Palindrom kilde.txt resultat.txt
```

der Palindrom er navnet på applikasjonen, kilde.txt er filen som det skal søkes i, og resultat.txt er filen eventuelle palindrome setninger skal skrives til. (Brukeren skal kunne velge filnavn fritt, "kilde.txt" og "resultat.txt" er bare eksempler.)

Slutt på oppgaveteksten



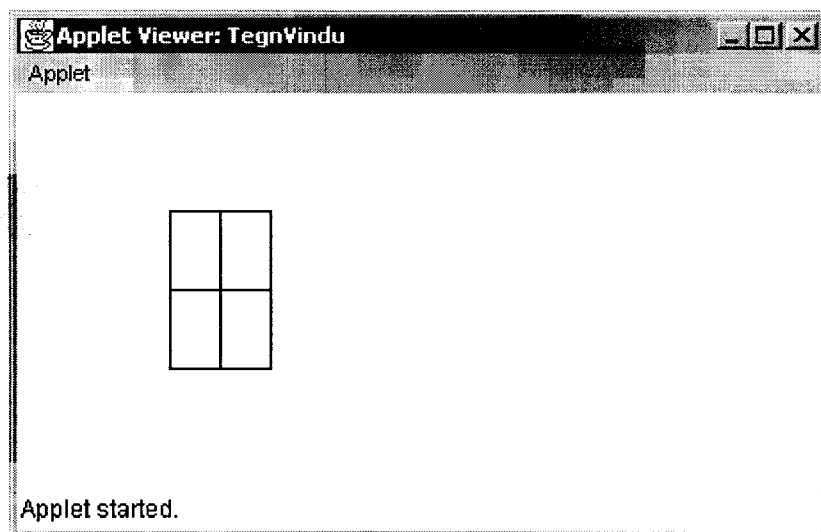
Oppgåvesettet inneheld tre uavhengige delar. Prosentsatsen ut for kvar oppgåve tyder veking ved sensur. Alle deloppgåver i ei gitt oppgåve har lik vekt. Prøv å svare på så mange oppgåver som mulig. Lukke til!

Oppgåve 1: Grafikk og metodar (40%)

I denne oppgåva skal du skrive *metodar*, ikkje komplette program.

1-a)

Lag ein metode *tegnVindu* som teiknar eit vindauge med to srosser, som i følgjande figur:

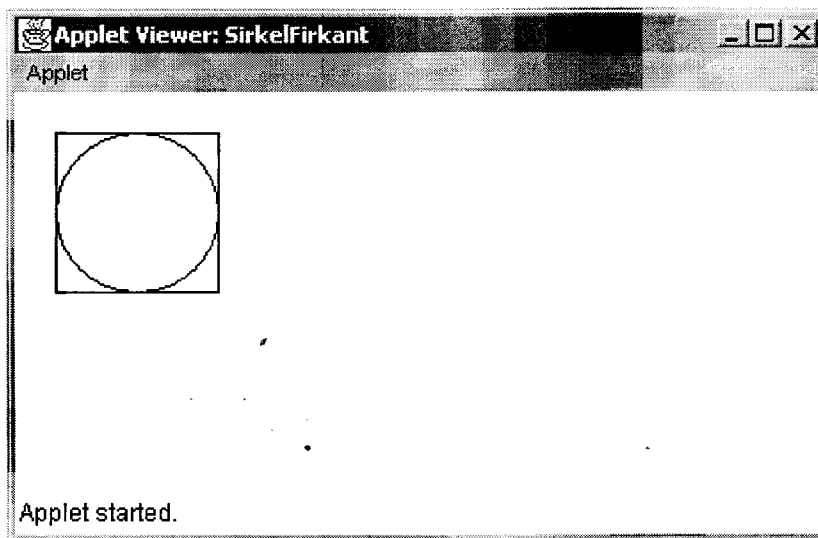


```
public void tegnVindu( Graphics g, int midtX, int midtY, int
bredde, int høyde){
    // Skriv kode her.
}
```

Her er midtX og midtY koordinatane til punktet der srossene møtes midt i vindauget, og bredde og høyde er vindaugets breidde og høgde.

1-b)

Lag ein metode *tegnSirkelFirkant* som teiknar ein sirkel omslutta av ein firkant, som i følgjande figur:

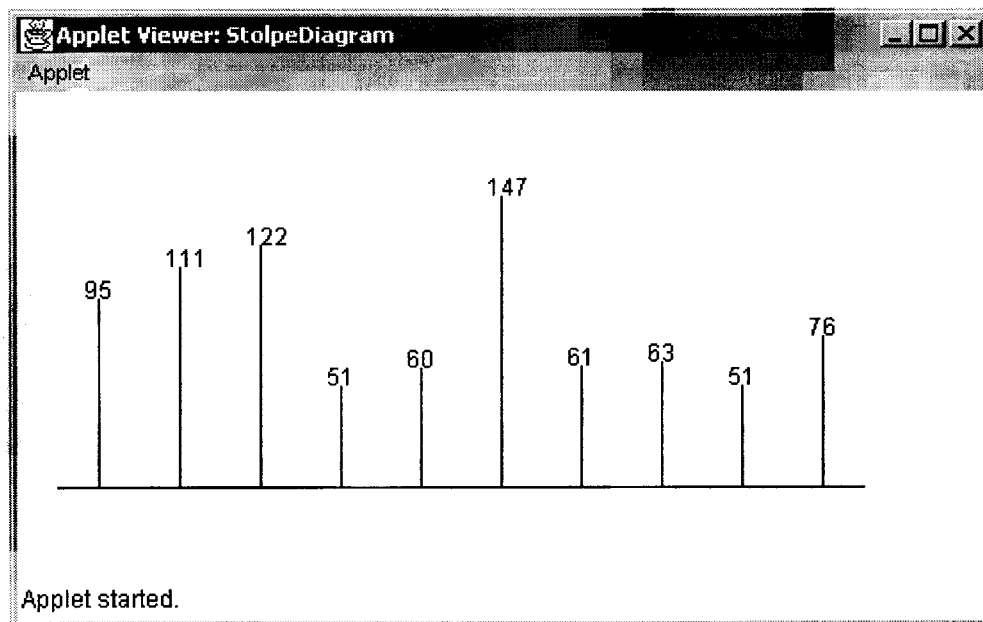


```
public void tegnSirkelFirkant ( Graphics g, int midtX, int  
midtY, int radius) {  
    // Skriv kode her.  
}
```

Her er midtX, midtY koordinatane til midtpunktet i sirkelen, og radius er radien til sirkelen.

1-c)

Lag ein metode *tegnStolpeDiagram* som teiknar eit stolpediagram, som i følgjande figur:



```
public void tegnStolpediagram( Graphics g, int maxY, int antallStolper ){  
    // Skriv kode her.  
}
```



Den første stolpen teiknas 40 pikslar inn i x-retning. Avstanden mellom kvar stolpe er 40 pikslar, høgda til kvar stolpe er eit tilfeldig tal mellom grensene 0 og maxY. Metoden genererer sjølv stolpehøgda (tala i figuren over er bare eit eksempel). Det skal teiknas antallStolper stolpar. Teikn x-aksen slik at den startar 20 pikslar inn og (maxY +50) pikslar ned i appletvindauget, og har ei lengde som er 40 ganger talet på stolpar.

1-d)

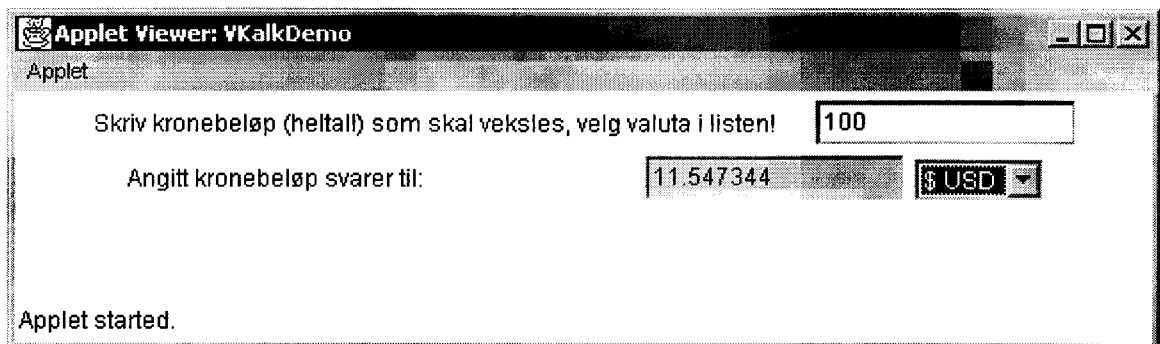
Skriv ein metode *fillMedPartall* som oppretter, initialiserer og returnerer ein tabell. Tabellen skal vere fylt med alle partala (i stigande rekkefølge) frå 2 til det partalet som er mindre enn eller lik parameteren n.

Eksempel: metodekallet `fillMedPartall(9)` returnerer ein referanse til tabellen {2, 4, 6, 8}.

```
public int[] fillMedPartall ( int n ) {  
    // Skriv kode her.  
}
```

Oppgåve 2: Valutakalkulator (30%)

I denne oppgåva skal du lage ein applett med tilhøyrande modellklasse (model-view-controller). Appletten skal ha følgjande brukargrensesnitt:



- Brukaren skal skrive inn eit kronebeløp i eit tekstfelt (bare heiltal), og velje ein valuta frå ei dropdown liste (klassa Choice).
- Programmet skal da rekne ut kva kronebeløpet svarar til i valt valuta, og skrive resultatet i eit tekstfelt, som i figuren.

Tips: Klassa Choice har ein metode `public int getSelectedIndex()` som returnerer indeksen til det valde elementet i Choice-lista.

For å gjere det enklare er variablane *valuta* og *valutakurs* gitt direkte i modellklassa VKalk. Ein spesifikasjon av klassa finn du under. Det er IKKJE forklart kva dei ulike metodane skal gjere, det må du sjølv finne ut.

```
public class VKalk{
```



```
private String[] valuta = {"NOK", "£ GBP", "$ USD", "Euro", "Yen"};
private float[] valutaKurs = {1.0f, 12.447f, 8.660f, 7.623f, 6.556f };
// andre instansvariabler i VKalk

public VKalk(){
    // kode for konstruktør
}
public int getValutaAntall(){
    // Skriv kode her.
}
public String getValuta( int index){
    // Skriv kode her.
}
public void setBeløp( float Beløp ){
    // Skriv kode her.
}

public float getValutaBeløp(){
    // Skriv kode her.
}

public void setValutaIndex( int index){
    // Skriv kode her.
}

// Om du treng hjelpemetodar kan du lage det
}
```

2a)

Skriv kode for appleten VKalkDemo. Appleten skal bruke metoder i modellklassa VKalk. Det kan lønne seg å studere modellklassa godt før du begynner å programmere.

2b)

Bestem nødvendige instansvariabler og skriv kode til alle metodane i modellklassa VKalk.

Oppgave 3: Ikke-grafisk applikasjon med filbehandling (30%)

Ei palindrom setning er ei setning der bokstavanes rekkefølge er den same når setninga blir lest både framlengs og baklengs. Alle teikn som ikkje er bokstaver (komma, blanke, utropsteikn, spørsmålsteikn osv) skal bli ignorert i denne samanhengen. Om det er store eller små bokstavar blir og ignorert.

Her er noen eksempel (på engelsk):

Dammit, I'm mad!

A nut for a jar of tuna.

A santa lived as a devil at NASA.

A Toyota.



Flee to me, remote elf!
GNU dung.
Kayak.
Name no one man,
"Naomi", I moan.
Yawn--Madonna fan? No Damn way!

Lag ein ikkje-grafisk applikasjon som leitar gjennom ein tekstfil etter setningar som er palindrome. Kvar setning står på ei eiga linje. Programmet lagar ei ny fil og skriv eventuelle palindrome setningar til denne nye fila.

Her er ein hjelpemetode du kan nytte i løysinga di:

```
private String fjernSpesialTegn( String linje){  
    String bokstaver = "";  
    for (int i = 0; i < linje.length(); i++){  
        char tegn = linje.charAt(i);  
        if (Character.isLetter(tegn) )  
            bokstaver+=tegn;  
    }  
    return bokstaver;  
}
```

Programmet skal nyttast slik:

```
C:\ java Palindrom kilde.txt resultat.txt
```

der Palindrom er namnet på applikasjonen, kilde.txt er fila som skal søkast i, og resultat.txt er fila eventuelle palindrome setningar skal skrivast til. (Brukaren skal kunne velje filnamn fritt, "kilde.txt" og "resultat.txt" er bare eksempel.)

Slutt på oppgåveteksten